Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers and Mathematics with Applications 49 (2005) 239–252
An International Journal
computers & mathematics with applications
www.elsevier.com/locate/camwa

A Second-Order Scheme for Integration of One-Dimensional Dynamic Analysis

HANG MA
Department of Mechanics, College of Sciences
Shanghai University, Shanghai 200436, P.R. China
QING-HUA QIN
Department of Engineering, Australian National University
ACT 0200, Australia

(Received March 2004; revised and accepted August 2004)

Abstract—This paper proposes a second-order scheme of precision integration for dynamic analysis with respect to long-term integration. Rather than transforming into first-order equations, a recursive scheme is presented in detail for direct solution of the homogeneous part of second-order algebraic and differential equations. The sine and cosine matrices involved in the scheme are calculated using the so-called 2^N algorithm. Numerical tests show that both the efficiency and the accuracy of homogeneous equations can be improved considerably with the second-order scheme. The corresponding particular solution is also presented, incorporated with the second-order scheme where the excitation vector is approximated by the truncated Taylor series.

Keywords—Precision integration, Second-order scheme, Initial problem, Differential quadrature method.

1. INTRODUCTION

Analysis of the dynamic responses of structures is of great engineering importance in practice. These responses are controlled by second-order differential equations, which are traditionally analyzed by means of direct integration or step-by-step schemes, among which are implicit methods, such as the Newmark, Wilson-θ, or Houbolt schemes, and explicit methods such as the central difference scheme. Recently, a high precision integration scheme was proposed by Zhong and Williams [1,2] for structural dynamics. Here the term 'precision integration' is specially referred as to a class of time integration procedure [1–7] which uses a recurrence formula to reduce the computing effort and simplifies the exponential matrix method. It is a numerical rather than an analytical one with which the machine precision can be achieved for the solution of homogeneous part on ordinary computers. The high precision integration is an unconditionally stable explicit method, which has been described in several research publications, exhibiting high precision and efficiency in comparison with traditional methods, such as that of Newmark [3–7]. However, these

This work was financially supported by the Australian Research Council. This support is gratefully acknowledged.

0898-1221/05/$ - see front matter Typeset by AMS-TEX
doi:10.1016/j.camwa.2004.08.009

high precision integration schemes are based on a system of first-order differential equations, so that second-order differential equations need to be transformed into first-order equations before the numerical solution can be performed. This would make the system matrix large and the scheme less efficient. In the present work, precision integration is extended to direct solution of the second-order algebraic and differential equations, whereby efficiency and accuracy can be further improved. This is considered significant especially for dynamic analysis with an emphasis on long-term dynamic evolution. The sine and cosine matrices involved in the second-order scheme are calculated using the so-called 2^N algorithm. The corresponding particular solution is also presented incorporated with the second-order scheme, where the excitation vector is approximated by the truncated Taylor series. Numerical tests show that both the efficiency and the accuracy of the homogeneous solution can be improved considerably with the proposed second-order scheme.

2. FORMULATION OF THE HIGH PRECISION INTEGRATION SCHEME

The equation of motion of a discretized structural model can be written as a second-order algebraic and differential equation in matrix form as

$$\mathbf{M}\ddot{\mathbf{u}} + \mathbf{B}\dot{\mathbf{u}} + \mathbf{K}\mathbf{u} = \mathbf{f}(t), \tag{1}$$

with initial conditions given as

$$\mathbf{u}(t_0) = \mathbf{u_0}, \qquad \dot{\mathbf{u}}(t_0) = \dot{\mathbf{u}}_0, \tag{2}$$

where $\mathbf{M}$, $\mathbf{B}$, and $\mathbf{K}$ represent time-invariant mass, damping and stiffness matrices with order n of the structure, respectively, with $\mathbf{M}$ being assumed positively definite. $\mathbf{f}(t)$ is the excitation vector. The dot over a variable indicates differentiation of the state variable with respect to the time t.

2.1. The First-Order Scheme

In the first-order scheme, by introducing the new variable $\mathbf{v} = \{\mathbf{u}^\top, (\mathbf{M}\mathbf{u} + \mathbf{B}\dot{\mathbf{u}}/2)^\top\}^\top$, equation (1) is first transformed into a form of first-order algebraic and differential equations as

$$\dot{\mathbf{v}} = \mathbf{H}\mathbf{v} + \mathbf{r}, \tag{3}$$

where

$$r = \begin{Bmatrix} \mathbf{0} \\ \mathbf{f} \end{Bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} \dfrac{-\mathbf{M}^{-1}\mathbf{B}}{2} & \mathbf{M}^{-1} \\ \dfrac{\mathbf{B}\mathbf{M}^{-1}\mathbf{B}}{4} - \mathbf{K} & \dfrac{-\mathbf{B}\mathbf{M}^{-1}}{2} \end{bmatrix}. \tag{4}$$

Since $\mathbf{H}$ is a constant matrix, from the theory of ordinary differential equations, the homogeneous solution of equation (3) can be expressed as

$$\mathbf{v}_h = \exp\left[\mathbf{H}(t - t_0)\right]\mathbf{c}_0, \tag{5}$$

where $\mathbf{c}_0$ is a constant vector to be determined by initial conditions. Let $\mathbf{v}_p$ be the particular integral of the inhomogeneous term, then the general solution of equation (3) has the form of

$$\mathbf{v} = \mathbf{v}_h + \mathbf{v}_p = \exp\left[\mathbf{H}(t - t_0)\right]\ \mathbf{c}_0 + \mathbf{v}_p. \tag{6}$$

The vector $\mathbf{c}_0$ can then be readily determined from equation (6) by taking $t = t_0$ as $\mathbf{c}_0 = \mathbf{v}(t_0) - \mathbf{v}_p(t_0)$, therefore

$$\mathbf{v} = \exp\left[\mathbf{H}(t - t_0)\right]\left[\mathbf{v}(t_0) - \mathbf{v}_p(t_0)\right] + \mathbf{v}_p. \tag{7}$$

Letting Δt be the constant time step size, then we can write the explicit recursive formula for the general solution at $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\Delta t$, as

$$\mathbf{v}_{k+1} = \exp(\mathbf{H}\Delta t)\left[\mathbf{v}_k - \mathbf{v}_{p,k}\right] + \mathbf{v}_{p,k+1}, \qquad (k = 0, 1, \ldots). \tag{8}$$

The key point of the precision integration is the calculation of $\exp(\mathbf{H}\Delta t)$ as accurately as possible [1,2] as the accuracy of equation (8) depends on how accurately the exponential matrix function $\exp(\mathbf{H}\Delta t)$ can be evaluated. Because of its wide application, the calculation of the matrix exponential has been discussed in the literature [8]. Zhong and Williams [1,2] proposed an accurate and efficient method for the calculation of $\exp(\mathbf{H}\Delta t)$ also called 2^N algorithm, which has the form

$$\exp(\mathbf{H}\Delta t) = \left[\exp(\mathbf{H}\tau)\right]^m, \tag{9}$$

where $\tau = \Delta t/m$, $m = 2^N$, and N is a positive integer. It has been argued [2] that precision integration can achieve machine accuracy because τ is, in general, very small, even for a small value of N. For example, if $N = 20$ then $\tau = \Delta t/1048576$. In view of structural dynamics, this value of τ is much smaller than the highest modal period of any conventional idealized structure. However, this first-order scheme will make the system matrix relatively large. For example, if the size of $\mathbf{M}$, $\mathbf{B}$, and $\mathbf{K}$ in equation (1) is $n \times n$, then the size of $\mathbf{H}$ becomes $2n \times 2n$. Therefore, the efficiency of the method can be further improved, especially in the case of long term integration, by introducing the second-order scheme described in the following subsection.

2.2. The Second-Order Scheme

We now discuss in detail the second-order scheme for the homogeneous solution of equation (1) by assuming $\mathbf{f}(t) = \mathbf{0}$ as follows:

$$\mathbf{M}\ddot{\mathbf{u}}_h + \mathbf{B}\dot{\mathbf{u}}_h + \mathbf{K}\mathbf{u}_h = \mathbf{0}, \tag{10}$$

with the initial conditions

$$\mathbf{u}_h(t_0) = \mathbf{u}_0, \qquad \dot{\mathbf{u}}_h(t_0) = \dot{\mathbf{u}}_0. \tag{11}$$

Suppose that the homogeneous solution has the form $\dot{\mathbf{u}}_h = \exp(\mathbf{A}t)\mathbf{a}$ and insert it into equation (10), where $\mathbf{a}$ represents an arbitrary vector. We have the characteristic matrix equation as

$$\mathbf{M}\mathbf{A}^2 + \mathbf{B}\mathbf{A} + \mathbf{K} = \mathbf{0}. \tag{12}$$

Solving (12), we get

$$\mathbf{A} = -\mathbf{D} \pm i\boldsymbol{\Omega}, \tag{13}$$

where i is the imaginary number ($i = \sqrt{-1}$). Matrices $\mathbf{D}$ and $\boldsymbol{\Omega}$ are expressed as follows, respectively:

$$\mathbf{D} = \frac{1}{2}\mathbf{M}^{-1}\mathbf{B} \tag{14}$$

$$\boldsymbol{\Omega} = \sqrt{\mathbf{J}} = \sqrt{\frac{\mathbf{M}^{-2}\mathbf{B}}{4} + \mathbf{M}^{-1}\mathbf{K}}. \tag{15}$$

Matrix Ω is the square root of matrix $\mathbf{J}$, which can be computed by the technique of matrix decomposition

$$\begin{aligned} \mathbf{J} &= \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{-1} = -\mathbf{Q}\left(i\sqrt{\boldsymbol{\Lambda}}\right)\mathbf{Q}^{-1}\mathbf{Q}\left(i\sqrt{\boldsymbol{\Lambda}}\right)\mathbf{Q}^{-1} = -(i\boldsymbol{\Omega})^2 = \boldsymbol{\Omega}^2, && (\lambda_k < 0), \\ \mathbf{J} &= \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{-1} = \mathbf{Q}\sqrt{\boldsymbol{\Lambda}}\mathbf{Q}^{-1}\mathbf{Q}\sqrt{\boldsymbol{\Lambda}}\mathbf{Q}^{-1} = \boldsymbol{\Omega}^2, && (\lambda_k > 0). \end{aligned} \tag{16}$$

That is

$$\mathbf{\Omega} = \mathbf{Q}\sqrt{\mathbf{\Lambda}}\mathbf{Q}^{-1}, \tag{17}$$

where $\mathbf{\Lambda}$ stands for the diagonal matrix which is composed of the eigenvalues of $\mathbf{J}$ as follows:

$$\mathbf{\Lambda} = \begin{bmatrix} |\lambda_1| & & \\ & \ddots & \\ & & |\lambda_n| \end{bmatrix} \tag{18}$$

and the k^{th} column in $\mathbf{Q}$ represents the eigenvector corresponding to the eigenvalue λ_k. The eigenvalues and eigenvectors of matrix $\mathbf{J}$ can be computed numerically using the subroutines described in [9]. Here an assumption has to be made that all the eigenvalues λ_k $(k = 1, \ldots, n)$ are distinct, real, and nonzero $(\lambda_k \neq 0)$. With the theory of ordinary differential equations [8,10], the homogeneous solution of equation (10) can be expressed as

$$\mathbf{u}_h = \exp\left[-\mathbf{D}\,(t - t_0)\right]\left[\cos\mathbf{\Omega}\,(t - t_0)\,\mathbf{c_1} + \sin\mathbf{\Omega}\,(t - t_0)\,\mathbf{c_2}\right], \tag{19}$$

where $\mathbf{c}_1$ and $\mathbf{c}_2$ represent constant vectors to be determined by initial conditions (11). Therefore, we have

$$\mathbf{u}_h = \exp\left[-\mathbf{D}\,(t-t_0)\right]\left\{\left[\cos\mathbf{\Omega}\,(t-t_0)+\mathbf{\Omega}^{-1}\mathbf{D}\sin\mathbf{\Omega}\,(t-t_0)\right]\mathbf{u}_0+\mathbf{\Omega}^{-1}\sin\mathbf{\Omega}\,(t-t_0)\,\dot{\mathbf{u}}_0\right\}, \tag{20}$$

$$\begin{aligned}\dot{\mathbf{u}}_h = &-\exp\left[-\mathbf{D}\,(t - t_0)\right]\left\{\left(\mathbf{\Omega} + \mathbf{\Omega}^{-1}\mathbf{D}^2\right)\sin\mathbf{\Omega}\,(t - t_0)\,\mathbf{u}_0\right. \\ &\left.+\left[\mathbf{\Omega}^{-1}\mathbf{D}\sin\mathbf{\Omega}\,(t - t_0) - \cos\mathbf{\Omega}\,(t - t_0)\right]\dot{\mathbf{u}}_0\right\}.\end{aligned} \tag{21}$$

Taking the notations $\mathbf{C} = \cos(\mathbf{\Omega}\Delta t)$, $\mathbf{S} = \sin(\mathbf{\Omega}\Delta t)$, and $\mathbf{E} = \exp(-\mathbf{D}\Delta t)$ and letting Δt be the length of a constant time step, then the explicit recursive formula for the homogeneous solution at the $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\Delta t$, can be written as

$$\mathbf{u}_{h,k+1} = \mathbf{E}\left[\left(\mathbf{C} + \mathbf{\Omega}^{-1}\mathbf{DS}\right)\mathbf{u}_{h,k} + \mathbf{\Omega}^{-1}\mathbf{S}\dot{\mathbf{u}}_{h,k}\right], \tag{22}$$

$$\dot{\mathbf{u}}_{h,k+1} = -\mathbf{E}\left[\left(\mathbf{\Omega} + \mathbf{\Omega}^{-1}\mathbf{D}^2\right)\mathbf{S}\mathbf{u}_{h,k} + \left(\mathbf{\Omega}^{-1}\mathbf{DS} - \mathbf{C}\right)\dot{\mathbf{u}}_{h,k}\right]. \tag{23}$$

The method of precise calculation of the exponential matrix $\mathbf{E}$ was proposed by Zhong and Williams [1,2]. Here we give the method for the precise calculation of the matrix cosine and sine functions $\mathbf{C}$ and $\mathbf{S}$. This is obviously the key point of the scheme. We start from the following exponential relation by letting $\tau = \Delta t/m$ and $m = 2^N$

$$\exp\left(i\mathbf{\Omega}\Delta t\right) = \left\{\exp\left(i\mathbf{\Omega}\tau\right)\right\}^m. \tag{24}$$

Expand the right-hand side of (24) to the form of Taylor series [8]

$$\exp\left(i\mathbf{\Omega}\tau\right) = \sum_{k=0}^{\infty}\frac{1}{k!}\left(i\mathbf{\Omega}\tau\right)^k. \tag{25}$$

As mentioned in the previous section, in general τ is extremely small, and so the series can be truncated to retain limited terms as follows:

$$\exp\left(i\mathbf{\Omega}\tau\right) \approx \mathbf{I} + \mathbf{T}_0 + i\mathbf{F}_0, \tag{26}$$

where $\mathbf{I}$ refers the identity matrix of order n and

$$\mathbf{T}_0 = -\frac{1}{2}\left(\mathbf{\Omega}\tau\right)^2 + \frac{1}{24}\left(\mathbf{\Omega}\tau\right)^4 - \frac{1}{720}\left(\mathbf{\Omega}\tau\right)^6, \tag{27}$$

$$\mathbf{F}_0 = \mathbf{\Omega}\tau - \frac{1}{6}\left(\mathbf{\Omega}\tau\right)^3 + \frac{1}{120}\left(\mathbf{\Omega}\tau\right)^5. \tag{28}$$

With the following relations

$$(\mathbf{I}+\mathbf{T}_k+i\mathbf{F}_k)^2=\mathbf{I}+\left(\mathbf{T}_k^2-\mathbf{F}_k^2+2\mathbf{T}_k\right)+2i\mathbf{F}_k\left(\mathbf{I}+\mathbf{T}_k\right)=\mathbf{I}+\mathbf{T}_{k+1}+i\mathbf{F}_{k+1},\quad (k=0,1,\ldots,N), \tag{29}$$

we can compute matrices $\mathbf{T}_N$ and $\mathbf{F}_N$ recursively. In FORTRAN language, the loop statement for the computation looks like this:

$$\text{do } k=0,N-1;\ \mathbf{T}_{k+1}=\mathbf{T}_k^2-\mathbf{F}_k^2+2\mathbf{T}_k;\ \mathbf{F}_{k+1}=2\mathbf{F}_k\left(\mathbf{I}+\mathbf{T}_k\right);\ \text{end do}$$

Invoking the Euler formula $\exp(i\boldsymbol{\Omega}\Delta t)=\cos(\boldsymbol{\Omega}\Delta t)+i\sin(\boldsymbol{\Omega}\Delta t)$, we can write

$$\{\exp\left(i\boldsymbol{\Omega}\tau\right)\}^m=\mathbf{C}+i\mathbf{S}\approx\left(\mathbf{I}+\mathbf{T}_0+i\mathbf{F}_0\right)^m=\mathbf{I}+\mathbf{T}_N+i\mathbf{F}_N. \tag{30}$$

That is, $\mathbf{C}=\cos(\boldsymbol{\Omega}\Delta t)\approx\mathbf{I}+\mathbf{T}_N$ and $\mathbf{S}=\sin(\boldsymbol{\Omega}\Delta t)\approx\mathbf{F}_N$. The truncated error of the expansion equation (26) can be estimated as $(\boldsymbol{\Omega}\tau)^7/7!$. By substituting equation (17) into (25) then taking the first seven terms, we have

$$\exp\left(i\boldsymbol{\Omega}\tau\right)\approx\mathbf{Q}\left[\sum_{k=0}^{6}\frac{1}{k!}\left(i\sqrt{\boldsymbol{\Lambda}}\tau\right)^k\right]\mathbf{Q}^{-1}. \tag{31}$$

Thus the truncated error from the k^{th} eigensolution corresponding to λ_k would be $(\tau\sqrt{|\lambda_k|})^7/7!$. Suppose ε is the allowed truncation error, then we have $\Delta t\sqrt{|\lambda_k|}<2^N(7!\varepsilon)^{1/7}$. If we neglect the inherent damping, the eigenvalue $|\lambda_k|$ would in fact be the k^{th} angular frequency of the structure, that is, $|\lambda_k|=\omega_k$. It follows, by substituting $\omega_k=2\pi/T_k$, where T_k is the k^{th} natural period of the structure, that

$$\frac{\Delta t}{\sqrt{T_k}}<\frac{2^N\left(7!\varepsilon\right)^{1/7}}{\sqrt{2\pi}} \tag{32}$$

Since 2^N is, in general, a very large number, the time step size Δt can be assumed to be relatively large in numerical computation, even for an extremely small value of ε. If N is a moderate number which is not too small, the accuracy of the algorithm would not be dominated by the step size Δt in the sense of numerical computation. For example, suppose $\varepsilon=10^{-17}$, which reaches or exceeds computer precision, and $N=20$, then we have from (32) that $\Delta t/\sqrt{T_k}<2100$. This means that there would be no significant truncation error induced by using equation (26), even if the time step size is chosen to be 2100 times the square root of the k^{th} natural period of the structure. In practice, taking into consideration the effect of higher modes, the contribution of high modes to the solution would be damped out because of the inherent damping effect of the structure. It follows that if $N=20$ the precision integration stated above will give essentially the exact homogeneous solution. In other words, matrices $\mathbf{C}$ and $\mathbf{S}$, thus computed reflect the characteristics of the structure, including those of its higher mode. As a result of this, the accuracy of the general solution of equation (1) would be dominantly controlled by the accuracy of the particular solution.

2.3. The Particular Solution

If $\mathbf{f}(t)\neq\mathbf{0}$, the general solution of equation (1) consists of the homogeneous and the inhomogeneous or the particular solution which satisfies

$$\mathbf{M}\ddot{\mathbf{u}}_p+\mathbf{B}\dot{\mathbf{u}}_p+\mathbf{K}\mathbf{u}_p=\mathbf{f}\left(t\right). \tag{33}$$

The particular solution incorporated with the second-order scheme can be derived [10] with the aid of the homogeneous solution (19). Suppose

$$\mathbf{u}_p=\mathbf{U}_1\mathbf{k}_1+\mathbf{U}_2\mathbf{k}_2, \tag{34}$$

where $\mathbf{k}_1$ and $\mathbf{k}_2$ are vectors to be determined and

$$\begin{aligned} \mathbf{U}_1 &= \exp\left[-\mathbf{D}\left(t-t_0\right)\right]\cos\left[\mathbf{\Omega}\left(t-t_0\right)\right], \\ \mathbf{U}_2 &= \exp\left[-\mathbf{D}\left(t-t_0\right)\right]\sin\left[\mathbf{\Omega}\left(t-t_0\right)\right]. \end{aligned} \tag{35}$$

By solving the following matrix differential equations:

$$\begin{vmatrix} \mathbf{U}_1 & \mathbf{U}_2 \\ \dot{\mathbf{U}}_1 & \dot{\mathbf{U}}_2 \end{vmatrix} \begin{Bmatrix} \dot{\mathbf{k}}_1 \\ \dot{\mathbf{k}}_2 \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{M}^{-1}\mathbf{f} \end{Bmatrix}, \tag{36}$$

we can derive

$$\mathbf{k}_1 = \int_{t_0}^{t} \Delta^{-1} \begin{vmatrix} \mathbf{0} & \mathbf{U}_2 \\ \mathbf{M}^{-1}\mathbf{f} & \dot{\mathbf{U}}_2 \end{vmatrix} \mathbf{f}\left(\eta\right) d\eta, \qquad \mathbf{k}_2 = \int_{t_0}^{t} \Delta^{-1} \begin{vmatrix} \mathbf{U}_1 & \mathbf{0} \\ \dot{\mathbf{U}}_2 & \mathbf{M}^{-1}\mathbf{f} \end{vmatrix} \mathbf{f}\left(\eta\right) d\eta, \tag{37}$$

where

$$\Delta = \begin{vmatrix} \mathbf{U}_1 & \mathbf{U}_2 \\ \dot{\mathbf{U}}_1 & \dot{\mathbf{U}}_2 \end{vmatrix} = \mathbf{\Omega}\exp\left[-2\mathbf{D}\left(t-t_0\right)\right]. \tag{38}$$

Combining equations (34)–(38), we can write

$$\mathbf{u}_p = \mathbf{M}^{-1}\mathbf{\Omega}^{-1}\int_{t_0}^{t} \exp\left[-\mathbf{D}\left(t-\eta\right)\right]\sin\mathbf{\Omega}\left(t-\eta\right)\mathbf{f}\left(\eta\right) d\eta, \tag{39}$$

$$\dot{\mathbf{u}}_p = \mathbf{M}^{-1}\int_{t_0}^{t} \exp\left[-\mathbf{D}\left(t-\eta\right)\right]\left[\cos\mathbf{\Omega}\left(t-\eta\right) - \mathbf{\Omega}^{-1}\sin\mathbf{\Omega}\left(t-\eta\right)\right]\mathbf{f}\left(\eta\right) d\eta. \tag{40}$$

By substituting the integral variable with $\tau = t - \eta$, the particular solution at the $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\,\Delta t$, can be written as

$$\mathbf{u}_{p,k+1} = \mathbf{M}^{-1}\mathbf{\Omega}^{-1}\int_{0}^{\Delta t} \exp\left(-\mathbf{D}\tau\right)\sin\left(\mathbf{\Omega}\tau\right)\mathbf{f}\left(t_k + \Delta t - \tau\right) d\tau, \tag{41}$$

$$\dot{\mathbf{u}}_{p,k+1} = \mathbf{M}^{-1}\int_{0}^{\Delta t} \exp\left(-\mathbf{D}\tau\right)\left[\cos\left(\mathbf{\Omega}\tau\right) - \mathbf{\Omega}^{-1}\sin\left(\mathbf{\Omega}\tau\right)\right]\mathbf{f}\left(t_k + \Delta t - \tau\right) d\tau. \tag{42}$$

Methods have been proposed [3–6] for the numerical treatment of particular solutions according to the properties of $\mathbf{f}(t)$. In the present work, however, suppose that $\mathbf{f}(t)$ are smooth functions within one time step, so we can compute the particular solution with the aid of Taylor expansion as

$$\mathbf{f}\left(t_k + \Delta t - \tau\right) = \sum_{m=0}^{\infty} \frac{\mathbf{c}_m^k}{m!}\left(\Delta t - \tau\right)^m, \tag{43}$$

where

$$\mathbf{c}_m^k = \mathbf{f}^{(m)}\left(t_k\right). \tag{44}$$

Define the integrals including the term $(\Delta t - \tau)^m$ in the Taylor series as

$$\mathbf{I}_S^m = \int_{0}^{\Delta t} \exp\left(-\mathbf{D}\tau\right)\sin\left(\mathbf{\Omega}\tau\right)\left(\Delta t - \tau\right)^m d\tau, \tag{45}$$

$$\mathbf{I}_C^m = \int_{0}^{\Delta t} \exp\left(-\mathbf{D}\tau\right)\cos\left(\mathbf{\Omega}\tau\right)\left(\Delta t - \tau\right)^m d\tau. \tag{46}$$

We can deduce the recurrence formula for the calculation of the above integrals as follows:

$$\mathbf{I}_S^0 = \left(\mathbf{D}^2 + \mathbf{\Omega}^2\right)^{-1}\left(\mathbf{\Omega} - \mathbf{DES} - \mathbf{\Omega EC}\right), \tag{47}$$

$$\mathbf{I}_C^0 = \left(\mathbf{D}^2 + \mathbf{\Omega}^2\right)^{-1}\left(\mathbf{D} + \mathbf{\Omega ES} - \mathbf{DEC}\right), \tag{48}$$

$$\mathbf{I}_S^m = m\left(\mathbf{D}^2 + \mathbf{\Omega}^2\right)^{-1}\left(\frac{1}{m}\mathbf{\Omega}\Delta t^m - \mathbf{D}\mathbf{I}_S^{m-1} - \mathbf{\Omega}\mathbf{I}_C^{m-1}\right), \tag{49}$$

$$\mathbf{I}_C^m = m\left(\mathbf{D}^2 + \mathbf{\Omega}^2\right)^{-1}\left(\frac{1}{m}\mathbf{D}\Delta t^m + \mathbf{\Omega}\mathbf{I}_S^{m-1} - \mathbf{D}\mathbf{I}_C^{m-1}\right), \qquad (m = 1, 2, \ldots). \tag{50}$$

By combining the homogeneous solutions (22),(23) and the particular solutions (41),(42), we have the general solutions of equation (1) using the second-order scheme as follows:

$$\begin{aligned}
\mathbf{u}_{k+1} &= \mathbf{E}\left[\mathbf{C}+\boldsymbol{\Omega}^{-1}\mathbf{S}\left(\mathbf{u}_k+\dot{\mathbf{u}}_k\right)\right]+\boldsymbol{\Omega}^{-1}\mathbf{M}^{-1}\int_0^{\Delta t}\exp\left(-\mathbf{D}\tau\right)\sin\left(\boldsymbol{\Omega}\tau\right)\mathbf{f}\left(t_k+\Delta t-\tau\right)\,d\tau, \quad (51)\\
\dot{\mathbf{u}}_{k+1} &= -\mathbf{E}\left[\left(\boldsymbol{\Omega}+\boldsymbol{\Omega}^{-1}\mathbf{D}^2\right)\mathbf{S}\mathbf{u}_k+\left(\boldsymbol{\Omega}^{-1}\mathbf{D}\mathbf{S}-\mathbf{C}\right)\dot{\mathbf{u}}_k\right]\\
&\quad+\mathbf{M}^{-1}\int_0^{\Delta t}\exp\left(-\mathbf{D}\tau\right)\left[\cos\left(\boldsymbol{\Omega}\tau\right)-\boldsymbol{\Omega}^{-1}\mathbf{D}\sin\left(\boldsymbol{\Omega}\tau\right)\right]\mathbf{f}\left(t_k+\Delta t-\tau\right)\,d\tau. \quad (52)
\end{aligned}$$

3. NUMERICAL EXAMPLES

In this section, several simple numerical examples of one-dimensional problem are presented to test the proposed second-order scheme. The eigenvalues and the corresponding eigenvectors of the system matrix are computed using the FORTRAN packages in reference [9]. $N = 20$ is used in the 2^N algorithm. The first example serves as a basic test using the second-order algorithm for the solution of a homogeneous equation, which is compared with the first-order algorithm. The next two examples are for the test of the effect of the inhomogeneous part and the damping term of the differential equation on the solutions with the proposed algorithm. The numerical results of the first three examples are presented by the maximum errors, which are compared with the theoretical solutions. The last two numerical examples are wave propagation and impact problem, respectively. Their solution has some spatial and temporal nonsmoothness which is for further verifying the algorithm.

3.1. One-Dimensional Wave Equation

Consider the vibration of a string clamped at both ends. The governing equation is a one-dimensional wave equation

$$\ddot{u}=\frac{\partial^2 u}{\partial x^2}, \qquad x\in[0,2\pi], \tag{53}$$

with the boundary and initial conditions

$$u(0,t)=u(2\pi,t)=0, \qquad u(x,0)=\sin(x), \qquad \dot{u}(x,0)=0, \tag{54}$$

where $u(x,t)$ is the transverse displacement of the string with unit amplitude of initial displacement. In equation (54), the first two relations stand for the fixed boundary condition and the last two relations represent the initial condition. The solution to equation (53) is given by the D'Alembert integral, being of the form

$$u(x,t)=\sin(x)\cos(t). \tag{55}$$

We now discretize the spatial coordinate in equation (53) with the differential quadrature method [11,12]. In this method, the derivatives of $u(x,t)$ with respect to the spatial coordinate x are approximated by a weighted sum of function values at all discrete points within the interval $x\in[0,2\pi]$ under consideration, i.e.,

$$u^{(r)}(x_j,t)=\left.\frac{d^r}{dx^r}u(x,t)\right|_{x=x_j}=\sum_{k=0}^{n}A_{jk}^{(r)}u(x_k,t), \qquad (j=0,1,\ldots,n+1). \tag{56}$$

For the details of computation of weighting coefficients $A_{jk}^{(r)}$ $(r=2)$ refer to [13,14]. In the quadrature formula (56), $j=0$ and $j=n+1$ correspond to the boundary nodes, respectively, at $x=0$ and $x=2\pi$ so that n is the number of the interior nodes. The positions of the interior grid points are placed at Gaussian points, which are computed by the subroutine presented in

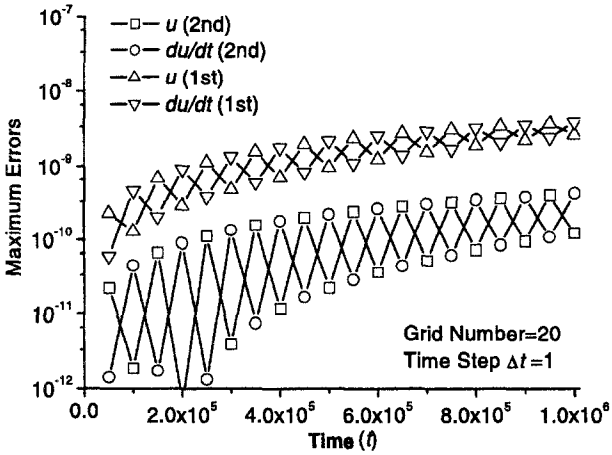

Figure 1. Comparison between the first- and second-order algorithms.

reference [15]. After discretization incorporated with the boundary condition, equation (53) is transformed to the following form of algebraic and differential equation as

$$\ddot{\mathbf{u}} + \mathbf{A}\mathbf{u} = \mathbf{0} \qquad \text{or} \qquad \ddot{\mathbf{u}} + \mathbf{\Omega}^2\mathbf{u} = \mathbf{0}, \tag{57}$$

where $\mathbf{\Omega} = \sqrt{\mathbf{A}}$. The numerical solution of the wave equation with the second-order precision integration algorithm at $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\Delta t$, is written as follows:

$$\mathbf{u}_{k+1} = \mathbf{C}\mathbf{u}_k + \mathbf{\Omega}^{-1}\mathbf{S}\dot{\mathbf{u}}_k, \tag{58}$$

$$\dot{\mathbf{u}}_{k+1} = -\mathbf{\Omega}\mathbf{S}\mathbf{u}_k + \mathbf{C}\dot{\mathbf{u}}_k. \tag{59}$$

The maximum solution errors using both the first- and second-order algorithms are compared in Figure 1. Twenty interior grid points and the time step $\Delta t = 1$ were used in the computation. It can be seen that both the algorithms are very stable, but better accuracy is achieved with the second-order algorithm. The CPU time used for the second-order algorithm required only about 37.5% of that used for the first-order algorithm for the same amount of computation. This is considered to result mainly from the small size of the system matrix in the second-order algorithm during long-term computation, because the major computational effort comes from the step-by-step integration realized by matrix multiplications.

The effect of the time step Δt on the accuracy with the second-order algorithm is shown in Figure 2. It can be seen that very large time steps can be employed for the solution of homogeneous equations with the precision integration scheme which verifies the error analysis presented in the Section 2.2.

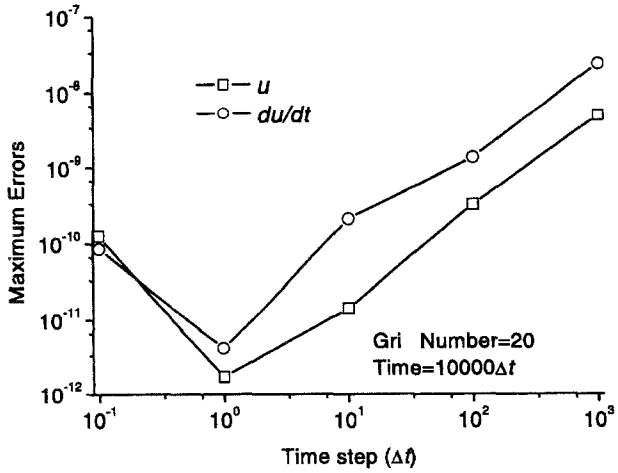

Figure 2. Errors as function of time step Δt with the second-order algorithm.

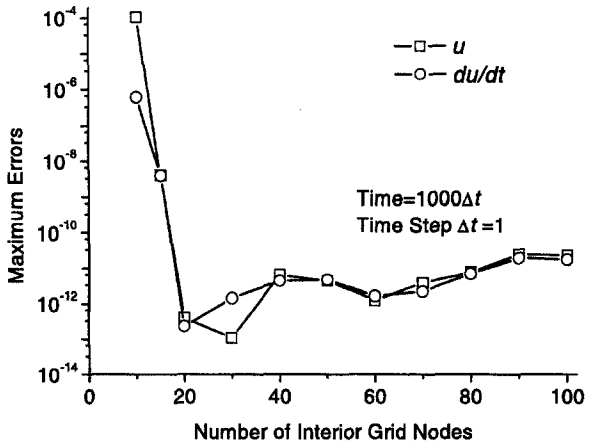

Figure 3. Errors as function of interior grid numbers with the second-order algorithm.

The effect of the interior grid number on the accuracy is shown in Figure 3. In reference [16], the wave equation was solved, accompanied by the differential quadrature discretization, by using the fourth-order Runge-Kutta method with a time step $\Delta t = 10^{-4}$. The solution in [16] becomes unstable at $t = 0.45$ even though the grid number is not too large ($n = 15$) in global discretization. In contrast, the results from the proposed algorithm show that no instability phenomenon occurs with the increase of the grid number after long-term integration ($t = 1000$) as shown in Figure 3. This fact demonstrates the superior feature of the precision integral compared to the traditional finite-difference-based time-marching method.

3.2. Forced One-Dimensional Wave Equation

Let us consider the forced vibration of a string clamped at both ends. The governing equation is the following inhomogeneous one-dimensional wave equation

$$\ddot{u} = \frac{\partial^2 u}{\partial x^2} + f(x,t), \tag{60}$$

where

$$f(x,t) = -3\sin(x)\sin(2t), \tag{61}$$

subject to the boundary and the initial conditions

$$u(0,t) = u(2\pi,t) = 0, \qquad u(x,0) = \sin(x), \qquad \dot{u}(x,0) = 2\sin(x). \tag{62}$$

The solution to equation (60) is as follows:

$$u(x,t) = \sin(x)\left[\cos(t) + \sin(2t)\right]. \tag{63}$$

After spatial discretization by the differential quadrature method, equation (60) becomes

$$\ddot{\mathbf{u}} + \mathbf{\Omega}^2\mathbf{u} = \mathbf{f}(t). \tag{64}$$

The numerical solution of the forced wave equation with the second-order precision integration algorithm at the $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\Delta t$, is written as follows:

$$\mathbf{u}_{k+1} = \mathbf{C}\mathbf{u}_k + \mathbf{\Omega}^{-1}\mathbf{S}\dot{\mathbf{u}}_k + \mathbf{\Omega}^{-1}\int_0^{\Delta t} \sin(\mathbf{\Omega}\tau)\,\mathbf{f}(t_k + \Delta t - \tau)\,d\tau, \tag{65}$$

$$\dot{\mathbf{u}}_{k+1} = -\mathbf{\Omega}\mathbf{S}\mathbf{u}_k + \mathbf{C}\dot{\mathbf{u}}_k + \int_0^{\Delta t} \cos(\mathbf{\Omega}\tau)\,\mathbf{f}(t_k + \Delta t - \tau)\,d\tau. \tag{66}$$

The integrals for the terms in the Taylor expansion of $\mathbf{f}(t)$ for the forced wave equation can be computed recursively by the following formula

$$\mathbf{I}_S^0 = \int_0^{\Delta t} \sin(\mathbf{\Omega}\tau)\, d\tau = \mathbf{\Omega}^{-1}(\mathbf{I} - \mathbf{C}), \tag{67}$$

$$\mathbf{I}_C^0 = \int_0^{\Delta t} \cos(\mathbf{\Omega}\tau)\, d\tau = \mathbf{\Omega}^{-1}\mathbf{S}, \tag{68}$$

$$\mathbf{I}_S^m = \int_0^{\Delta t} \sin(\mathbf{\Omega}\tau)(\Delta t - \tau)^m\, d\tau = m\mathbf{\Omega}^{-1}\left(\frac{1}{m}\mathbf{I}\Delta t^m - \mathbf{I}_C^{m-1}\right), \tag{69}$$

$$\mathbf{I}_C^m = \int_0^{\Delta t} \cos(\mathbf{\Omega}\tau)(\Delta t - \tau)^m\, d\tau = m\mathbf{\Omega}^{-1}\mathbf{I}_S^{m-1}, \qquad (m = 1, 2, \ldots). \tag{70}$$

The maximum errors as a function of time for the forced wave equation are shown in Figure 4. In the computation, only three terms in the Taylor expansion are sufficient to keep reasonable accuracy, and are retained. Figure 4 shows that the algorithm is long-term stable although the time step chosen should not be too large ($\Delta t = 0.1$) owing to the existence of the inhomogeneous term. By comparing Figure 4 with Figures 1–3 it is evident that the accuracy of the general solution is dominantly controlled by the accuracy of the particular solution.

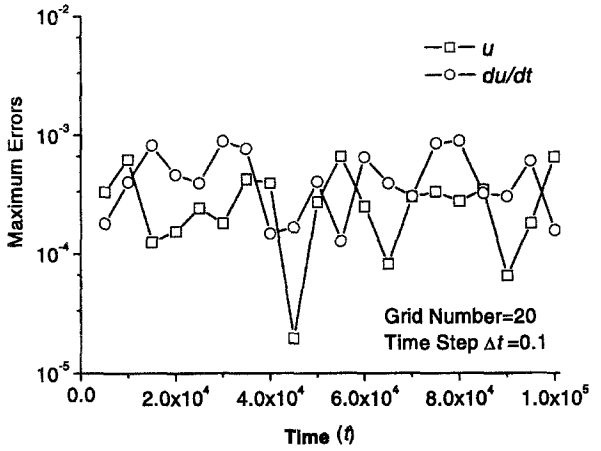

Figure 4. Errors as function of time for the forced wave equation.

3.3. Forced One-Dimensional Equation with Damping

The third example to be solved is a forced second-order one-dimensional equation with damping as follows:

$$\ddot{u} + 2\dot{u} + 3u = \frac{\partial^2 u}{\partial x^2} + f(x,t), \tag{71}$$

where

$$f(x,t) = 2\sin(x)\{2\cos(t) - \sin(2t)\} \tag{72}$$

subject to the boundary and initial conditions

$$u(0,t) = u(2\pi,t) = 0, \qquad u(x,0) = \sin(x), \qquad \dot{u}(x,0) = 3\sin(x). \tag{73}$$

The theoretical solution to equation (71) is as follows:

$$u(x,t) = \sin(x)[\cos(t)\exp(-t) + \sin(2t)]. \tag{74}$$

Equation (71) is transformed, after spatial discretization by the differential quadrature method, to the following form of algebraic and differential equation

$$\ddot{\mathbf{u}} + 2\dot{\mathbf{u}} + \mathbf{G}\mathbf{u} = \mathbf{f}(t), \tag{75}$$

where $\mathbf{G} = 3\mathbf{I} + \mathbf{A}$. The numerical solution of the forced equation with damping using the second-order precision integration algorithm at the $(k+1)^{\text{th}}$ time step, i.e., $t = t_0 + (k+1)\Delta t$, can be given as follows:

$$\mathbf{u}_{k+1} = e\left[\left(\mathbf{C} + \boldsymbol{\Omega}^{-1}\mathbf{S}\right)\mathbf{u}_k + \boldsymbol{\Omega}^{-1}\mathbf{S}\dot{\mathbf{u}}_k\right] + \boldsymbol{\Omega}^{-1}\int_0^{\Delta t} \exp(-\tau)\sin(\boldsymbol{\Omega}\tau)\,\mathbf{f}(t_k + \Delta t - \tau)\,d\tau, \tag{76}$$

$$\dot{\mathbf{u}}_{k+1} = -e\left[\left(\boldsymbol{\Omega} + \boldsymbol{\Omega}^{-1}\right)\mathbf{S}\mathbf{u}_k + \left(\boldsymbol{\Omega}^{-1}\mathbf{S} - \mathbf{C}\right)\dot{\mathbf{u}}_k\right] + \int_0^{\Delta t} \exp(-\tau)\left[\cos(\boldsymbol{\Omega}\tau) - \boldsymbol{\Omega}^{-1}\sin(\boldsymbol{\Omega}\tau)\right]\mathbf{f}(t_k + \Delta t - \tau)\,d\tau, \tag{77}$$

where $\boldsymbol{\Omega} = \sqrt{\mathbf{I} - \mathbf{G}}$ and $e = \exp(-\Delta t)$. The integrals of the terms in the Taylor expansion of $\mathbf{f}(t)$ for the forced equation with damping can also be computed recursively by the following formula:

$$\mathbf{I}_S^0 = \int_0^{\Delta t} \exp(-\tau)\sin(\boldsymbol{\Omega}\tau)\,d\tau = \mathbf{G}^{-1}(\boldsymbol{\Omega} - e\mathbf{S} - e\boldsymbol{\Omega}\mathbf{C}), \tag{78}$$

$$\mathbf{I}_C^0 = \int_0^{\Delta t} \exp(-\tau)\cos(\boldsymbol{\Omega}\tau)\,d\tau = \mathbf{G}^{-1}(\mathbf{I} + e\boldsymbol{\Omega}\mathbf{S} - e\mathbf{C}), \tag{79}$$

$$\mathbf{I}_S^m = \int_0^{\Delta t} \exp(-\tau)\sin(\boldsymbol{\Omega}\tau)(\Delta t - \tau)^m\,d\tau = m\mathbf{G}^{-1}\left(\frac{1}{m}\mathbf{I}\Delta t^m - \mathbf{I}_S^{m-1} - \boldsymbol{\Omega}\mathbf{I}_C^{m-1}\right), \tag{80}$$

$$\mathbf{I}_C^m = \int_0^{\Delta t} \exp(-\tau)\cos(\boldsymbol{\Omega}\tau)(\Delta t - \tau)^m\,d\tau = m\mathbf{G}^{-1}\left(\frac{1}{m}\mathbf{I}\Delta t^m + \boldsymbol{\Omega}\mathbf{I}_S^{m-1} - \mathbf{I}_C^{m-1}\right), \tag{81}$$

$$(m = 1, 2, \ldots).$$

The maximum errors as a function of time for the forced equation are shown in Figure 5. Again, only three terms in the Taylor expansion are retained in the computation. Figure 5 shows that this algorithm is also long-term stable. As the homogeneous part of the solution will decay to be negligibly small after a short period of time, the results in Figure 5 can be considered to be a contribution only by the inhomogeneous part.

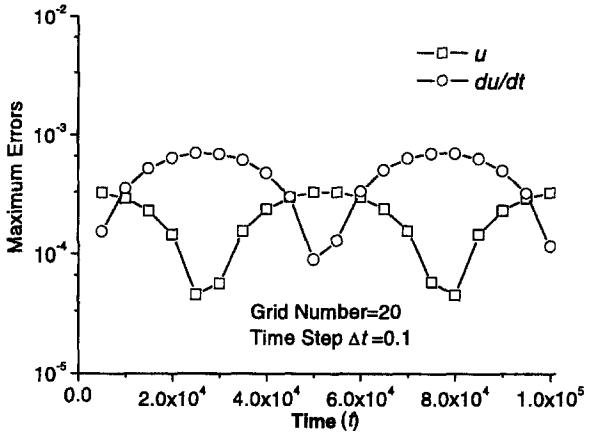

Figure 5. Errors as function of time for the forced equation with damping.

3.4. Wave Propagation of a String

The fourth example is the wave propagation in a string having the same governing equation of (53) but with different initial conditions. The initial wave profile is a localized packet, which will sweep over the whole string at subsequent time steps. In this case, the wave profile has a feature of some spatial nonsmoothness, a complicated problem as well as a good test for the algorithm. The boundary and initial conditions are as follows

$$u(0,t) = u(2\pi,t) = 0, \qquad \dot{u}(x,0) = 0,$$
$$u(x,0) = \begin{cases} \sin(2x - 1.5\pi), & (0.75\pi \le x \le 1.25\pi), \\ 0, & \text{other.} \end{cases} \tag{82}$$

It gives a localized wave concentrated on a central small area. In the numerical computations, the time step $\Delta t = 0.1$ and the grid number 100 were used. The wave profiles at selected time intervals are shown in Figure 6. It shows from Figure 6 that with the time increase, the wave propagates consecutively outwards towards the right and left ends, respectively. After hitting upon both ends, the waves are reflected as a negative wave and travels consecutively inwards towards the center. The two reflected waves meet again at center and will repeat the next cycle, which is not presented here in the figure. The results of this example also show that the proposed algorithm can provide very stable and accurate numerical solution.

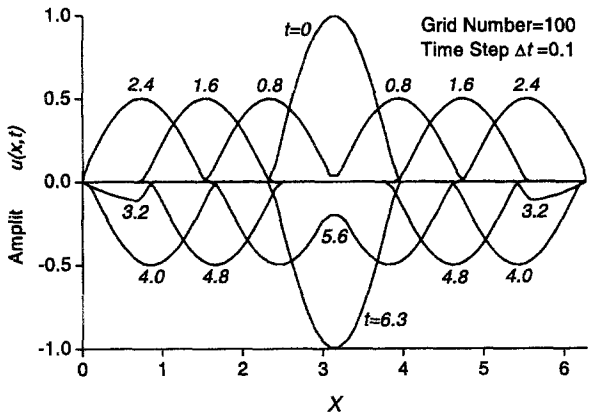

Figure 6. Wave propagation of a string.

3.5. Vibration of a Rod Subjected to a Heaviside-Type Impact

The final example is the vibration of a rod subjected to a Heaviside-type impact. In this case, the problem has the same governing equation of (53) but with different initial and boundary conditions as follows

$$u(x,0) = \dot{u}(x,0) = 0, \qquad u(1,t) = 0, \qquad \frac{\partial u}{\partial x}(0, t \ge 0) = 1. \tag{83}$$

It describes a Heaviside-type impact applied axially at one end of a rod ($x = 0,\ t \ge 0$) initially at rest. Its solution has a feature of both spatial and temporal sharper gradient. In the numerical computations, the time step $\Delta t = 0.1$ and the grid number 20 were used. Figure 7 displays the solutions of time-displacement curves at $x = 0$ and $x = 0.5$, respectively, for the longitudinal vibration of the rod subjected to the Heaviside-type impact. It can be seen from Figure 7 that results for the impact case are not as good as those of the previous four examples, which may attribute to the following two factors. The first reason is that the spatial discretization has been made by the differential quadrature method, which is a global-type discretization therefore is not too appropriate for the solution with sharper gradient [12]. The second reason may be the lack

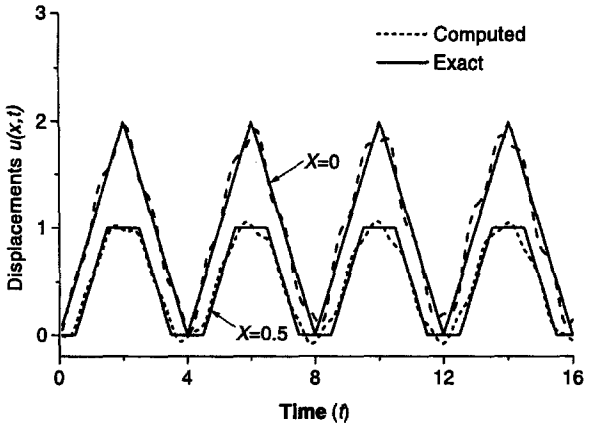

Figure 7. Displacement curves at $x = 0$ and $x = 0.5$ subjected to a Heaviside-type impact.

of numerical damping in the precise integration where the higher-order modes have important effect in the impact case.

The merit of the precise integration of the first-order is that it uses a recurrence formula to reduce the computing effort and simplifies the exponential matrix method. It can produce highly precise solutions yet keeping unconditionally stability for various dynamic problems [1–5] especially for smooth problems. The distinction of the second-order scheme from the first-order scheme lies in the treatment of the differential equations, which results in the sine and cosine matrices to be computed numerically as accurately as possible. The sizes of these matrices are much smaller than that of the exponential matrix. The major advantage of the proposed second-order scheme is that it keeps the main features of the first-order scheme while reduces the size of the matrix so that the computing effort can be further reduced during the step by step time integration which is especially important in long-term integration.

4. CONCLUSIONS

A second-order scheme of precision integration was proposed for dynamic analysis with respect to long-term integration. The recursive scheme is presented in detail for the direct solution of the homogeneous part of the second-order algebraic and differential equations. The corresponding particular solution is also presented, incorporated with the second-order scheme where the excitation vector is approximated by the truncated Taylor series. The sine and cosine matrices involved in the scheme are calculated using the so-called 2^N algorithm. Numerical tests show that both the efficiency and the accuracy of the homogeneous equation can be improved considerably with the second-order scheme especially for long-term integrations. The results also show the proposed algorithm is long-term stable even for a relatively larger time step.

REFERENCES

1. W.X. Zhong, On the precise time-integration method for structural dynamics, (in Chinese), *Journal of Dalian University of Technology* **34** (2), 131–136, (1994).
2. W.X. Zhong and F.W. Williams, A precise time step integration method, In *Proceedings of Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* **208**, 427–430, (1994).
3. J.H. Lin, W.P. Shen and F.W. Williams, A high precision direct integration scheme for structures subjected to transient dynamic loading, *Computers & Structures* **56**, 113–120, (1995).
4. J.H. Lin, W.P. Shen and F.W. Williams, A high precision direct integration scheme for non-stationary random seismic responses of non-classically damped structures, *Structural Engineering and Mechanics* **3**, 215–228, (1995).
5. J.H. Lin, W.P. Shen and F.W. Williams, Accurate high-speed computation of non-stationary random structural response, *Engineering Structures* **19**, 586–593, (1997).
6. X.J. Liu, D.W. Begg, M.A. Devane and W.X. Zhong, High precision integration for dynamic structural systems with holonomic constraints, *Structural Engineering and Mechanics* **5**, 283–295, (1997).

7. X.Q. Zhu and S.S. Law, Precise time-step integration for the dynamic response of a continuous beam under moving loads, *Journal of Sound and Vibration* **240**, 962–970, (2001).
8. A.S. Deif, *Advanced Matrix Theory for Scientists and Engineers*, Second Edition, Abacus Press, New York, (1991).
9. B.T. Smith, J.M. Boyle, B.S. Garbow, Y. Ikebe, V.C. Klema and C.B. Moler, *Matrix Eigensystem Routines-EISPACK Guide*, Springer-Verlag, Berlin, (1976).
10. P. Bugl, *Differential Equations-Matrices and Models*, Prentice Hall, New Jersey, (1995).
11. R.E. Bellman, B.G. Kashef and J. Casti, Differential quadrature: A technique for rapid solution of nonlinear partial differential equations, *Journal of Computational Physics* **10**, 40–52, (1972).
12. C.W. Bert and M. Malik, Differential quadrature method in computational mechanics: A review, *Applied Mechanics Review* **49**, 1–27, (1996).
13. J.R. Quan and C.T. Chang, New insight in solving distributed system equations by the quadrature method—I. Analysis, *Computers in Chemical Engineering* **13**, 779–788, (1989).
14. J.R. Quan and C.T. Chang, New insight in solving distributed system equations by the quadrature method—II. Numerical experiments, *Computers in Chemical Engineering* **13**, 1017–1024, (1989).
15. H.P. William, A.T. Saul, T.V. William and P.F. Brian, *Numerical Recipes in Fortran77, Volume 1: The Arts of Scientific Computing*, Cambridge University Press, Cambridge, (1997).
16. Z. Zong and K.Y. Lam, A localized differential quadrature (LDQ) method and its application to the 2D wave equation, *Computational Mechanics* **29**, 382–391, (2002).